What is claimed is:

1. In a device comprising a programmable processor with an associated physical memory space which stores native functions in a format determined by a hardware architecture of the processor, the improvement comprising a virtual machine concurrently emulated by the processor which executes virtual functions in a standardized format determined irrespective of the hardware architecture of the processor, wherein the execution of a selected virtual function by the virtual machine causes the processor to execute a corresponding native function, and wherein the corresponding native function is executed to evaluate concurrent execution of at least one other native function.

2. The improvement of claim 1, wherein the virtual machine comprises a symbol table that maps each virtual function to the corresponding native function.

3. The improvement of claim 2, wherein the virtual machine further comprises an execution block coupled to the symbol table which executes the virtual functions.

4. The improvement of claim 3, wherein the virtual machine further comprises a gate call interface block which, when the execution engine block executes the selected virtual function, accesses the symbol table and initiates a gate call function to cause the processor to execute the corresponding native function.

5. The improvement of claim 1, wherein the execution of the corresponding native function by the processor results in a return data value, and wherein the virtual machine further operates to provide the return data value to a host device.

6. The improvement of claim 1, wherein the native functions comprise operational routines and test routines, the operational routines carried out to

facilitate normal operation of the device and the test routines carried out to identify error conditions associated with the normal operation of the device.

7. The improvement of claim 1, wherein the processor is characterized as a first processor, and wherein the virtual machine is further emulated in a second processor having a different hardware architecture from the architecture of the first processor so that the same virtual functions are executed in both the first and second processors.

8. The improvement of claim 1, wherein the virtual machine further comprises a virtual memory register in which the virtual functions are loaded and sequentially executed.

9. The improvement of claim 1, wherein the device is characterized as a data storage device and the processor is characterized as a controller which provides top level communications and control functions for said device.

10. A method for concurrently emulating a virtual machine in a physical memory space of a programmable processor, the memory space storing native functions in a format determined by a hardware architecture of the processor, the method comprising:

providing a set of virtual functions each in a standardized format determined irrespective of the hardware architecture of the processor; and

using the virtual machine to execute a selected virtual function which causes the processor to execute a corresponding native function, wherein the corresponding native function evaluates concurrent execution of at least one other native function.

11. The method of claim 10, further comprising a prior step of generating a symbol table that maps selected virtual functions to the corresponding native functions.

12. The method of claim 11, wherein the using step further comprises accessing the symbol table to identify the corresponding native function and performing a gate call operation to execute the corresponding native function.

13. The method of claim 10, wherein the using step results in generation of a return data value, and wherein the method further comprises communicating the return data value to a host device.

14. The method of claim 10, wherein the processor is characterized as a first processor and wherein the method is performed in combination with a method for concurrently emulating a virtual machine in a second physical memory space of a second programmable processor, the second memory space storing native functions in a format determined by a second hardware architecture of the second processor different from the hardware architecture of the first processor, and wherein the providing and using steps are repeated for the second processor.

15. The method of claim 10, wherein the native functions comprise operational routines and test routines, the operational routines carried out to facilitate normal operation of the device and the test routines carried out to identify error conditions associated with the normal operation of the device.

16. The method of claim 10, wherein the processor is characterized as a controller in a data storage device.